REMARKS

This paper is responsive to Final Office Action dated October 6, 2004. Claims 9-36 were examined. All previous rejections have been withdrawn. However, all claims are now *newly* rejected as anticipated under 35 U.S.C. § 102(b) by U.S. Patent 5,193,167 to Sites et al (hereafter *Sites*). The rejections are traversed. In addition, the Examiner notes certain claim informalities which have been addressed per the Examiner's requirements.

Claim Objections

Each of the claim objections has been addressed, per Examiner's requirements, by a minor amendment to the objected-to claim.

Claim Rejections – 35 USC § 102

As a preliminary matter, Applicant thanks Examiner for withdrawal of the previously relied upon rejections under 35 U.S.C. § 102(b). We now turn to the present rejection.

Claims 9–36 were rejected under 35 U.S.C. 102(b) as being anticipated by Sites et al., U.S. Patent No. 5,193,167 (hereinafter *Sites*). In rejecting the pending claims, the Office simply misinterprets certain aspects of *Sites'* pipeline. In particular, the Office fails to properly identify any express or inherent disclosure of pipeline stages during which intra-group dependency checking and inter-group dependency checking are performed in accordance with the claim language.

Five errors of interpretation are significant:

(1) The Office apparently construes *Sites'* first four pipeline stages, i.e., fetch (S0), swap (S1), decode (S2) and register access/issue (S3) as "instruction grouping for dispatch, including both intra-group and inter-group dependency checking." *See* Final Action ¶12. In particular, the Office states that: "over multiple cycles (S0-S3) plus any stall cycles, dependencies and resource constraints are checked."

The Office's interpretation of Sites' initial pipeline stages through decode (i.e., fetch (S0), swap (S1) and decode (S2)) as dependency checking ignores the simple fact that decode (e.g., of register addresses) must have been completed to allow for checking of dependencies (intra-group or inter-group) between instructions. Accordingly, dependency checking is performed in Sites, if at all,

after decode (S2) in register access/issue (S3) stage. Of note, the register access/issue (S3) stage is a <u>single-cycle</u> stage.

(2) Related to (1), the Office's analysis seems to suggest (and perhaps relies upon reasoning that) a pipeline stall acts to transform a single-cycle operation into a multiple-cycle operation. *See* Final Action ¶12.

To the contrary, an operation that is performed in one cycle is still performed in one cycle whether or not, after performance of that operation, the pipeline is stalled for one or more subsequent cycles.

(3) The Office further construes *Sites'* first two pipeline stages, i.e., fetch (S0) and swap (S1), as intra-group dependency checking that spans multiple cycles. *See* Final Action, ¶16. To be specific, the Office states that: "fetching is considered a portion of intra-group dependency checking because if instructions aren't fetched, then they cannot be checked for dependencies.

Really? So, by the Office's reasoning: examination of a patent application must necessarily include both invention and application drafting since examination cannot be performed unless an invention is made and an application prepared. With utmost respect, Applicant notes that the Office's reasoning here is unsustainable.

(4) Next, and perhaps fundamentally, the Office assumes that Sites' swap stage (S1) constitutes intra-group dependency checking. *See* Final Action throughout, but particularly ¶¶16, 17.

No group yet exists at Sites' swap stage (S1) for which intra-group dependencies could be checked. Instead, Sites' swap stage (S1) simply determines whether two fetched instructions must be issued separately (i.e., in two distinct issue groups) or if the two fetched instructions may be issued at the same time, subject to dependency checking (presumably intra-group and inter-group dependency checking) performed at register access/issue stage (S3) after register addresses are decoded in decode stage (S2). In short, it is rather odd for the Office to take the position that intra-group dependency checking is performed in a stage that precedes the decoding of register addresses that would presumably be employed in the dependency checking.

(5) Finally, with respect to claim language relating to evaluation of "non-deterministic conditions," the Office leverages a definition of "nondeterminism" (Final Action, ¶¶19, 23, 33, 34) in a way that is simply absurd and contrary to accepted meaning.

While it is true that non-determinism is generally understood to be a property of a computation that may have more than one result, the Office neglects the additional and critical point that a nondeterministic computation is generally one that <u>for a given set of inputs</u> may have more than one result. Thus, addition is not non-deterministic because 2+2 and 2+3 yield different results. Similarly, intra-group dependency checking is not non-deterministic because one group of

instructions and register targets exhibits a dependency while another group of instructions and register targets does not.

Accordingly, in the context of the present application, a non-deterministic condition is one that, based on information available at a given time, cannot be precisely determined. The specification gives examples of future load buffer or store buffer state that (as implemented) are not precisely determinable.

Applicant respectfully suggests that the Office has misapplied its definition in a way that is nonsensical. To aid the Office, Applicants suggest the following definition may be less susceptible to misapplication: NONDETERMINISTIC-permitting more than one choice of next move at some step in a computation ***(From Algorithms and Theory of Computation Handbook, page 24-19, Copyright ©1999 by CRC Press LLC. Appearing in the Dictionary of Computer Science, Engineering and Technology, Copyright © 2000 CRC Press LLC; see also http://www.nist.gov/dads/HTML/nondeterministic.html)***

Specific Claim Rejections – All Pending Claims

Rejections of independent claims 1, 17, 26, 31 and 35 are all based on the errors of interpretation listed as (1) and (2) above. Properly interpreted, *Sites* does not disclose or suggest, taken alone or in combination, the subject matter of the independent claims, which respectively include the following limitations:

> … perform[ance], over plural execution cycles of the superscalar processor, instruction grouping for dispatch, including both intra-group and inter-group dependency checking, wherein the instruction grouping for dispatch takes the plural execution cycles to complete. (***Claim 1***)
>
> grouping logic coupled to the functional units and pipelined to compute, over plural cycles, T, of the processor, a future state, S(t+T), of the processor based on a prior state, S(t), of the processor and based thereon to select a group of instructions from a program sequence thereof for dispatch to the functional units, wherein the future state computing takes the plural cycles to complete. (***Claim 17***)
>
> performing, during plural pipelined execution cycles of the processor, dependency checking amongst instructions of a later one of the groups and between the instructions of the later group and instructions of a preceding one of the groups … wherein the performed dependency checking takes the plural pipelined execution cycles to complete. (***Claim 26***)

wherein, for instruction instances of the instruction group, dependency checking, which includes the intra-group dependency checking and the inter-group dependency checking, takes plural of the cycles of processor execution to complete. (***Claim 31***)

OR

means for grouping, over plural pipeline stages, instructions for dispatch to respective ones of the functional units, wherein the grouping of a particular set of instruction instances takes the plural pipeline stages to complete. (***Claim 35***)

For at least these reasons, claims 1, 17, 26, 31 and 35 and all those dependent therefrom (i.e., all pending claims) are allowable and a notice to that effect is respectfully requested.

Specific Claim Rejections – Claims 13 and 18

Rejections of dependent claims 13 and 18 are further based on the errors of interpretation listed as (3) above. Properly interpreted, *Sites* does not disclose or suggest, taken alone or in combination, intra-group dependency checking that spans at least two execution cycles. Claims 13 and 18 are allowable for at least this reason and a notice to that effect is respectfully requested.

Furthermore, at least the rejections of dependent claim 13 and 18 are still further based on the errors of interpretation listed as (4) above. Since the S1 stage of *Sites'* implementation does not constitute intra-group dependency checking, claims 13 and 18 are allowable for at least this reason and a notice to that effect is respectfully requested.

Specific Claim Rejections – Claims 16, 20, 21, 30, 33 and 34

Finally, all rejections of claims reciting the evaluation or checking of "non-deterministic conditions" or "non-deterministic dependency conditions" are flawed because of the errors of interpretation listed as (5) above. Properly interpreted, neither the express nor inherent description of *Sites* discloses or suggests, taken alone or in combination, evaluation of non-deterministic conditions as recited in independent claim 31 or as recited in dependent claims 16, 20, 30, 33 or 34. Claims 16, 20, 21, 30, 33 and 34 are therefore allowable and a notice to that effect is respectfully requested.

Conclusion

In summary, claims 9-36 are in the case. All claims are believed to be allowable over the art of record, and a Notice of Allowance to that effect is respectfully solicited. Nonetheless, if any issues remain that could be more efficiently handled by telephone, the Examiner is requested to call the undersigned at the number listed below.

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that, on the date shown below, this correspondence is being

☐ deposited with the US Postal Service with sufficient postage as first class mail, in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

☒ facsimile transmitted to the US Patent and Trademark Office.

David W. O'Brien — 7-Feb-05 (Date)

EXPRESS MAIL LABEL: _____

Respectfully submitted,

David W. O'Brien, Reg. No. 40,107
Attorney for Applicant(s)
(512) 338-6314
(512) 338-6301 (fax)